

Swarm-based Descriptor Combination and its Application for Image Classification

A. Mansano¹ and J. Mastuoka¹ and N. Abiuzzi¹ and L. Afonso¹ and J. Papa¹ and F. Faria² and R. Torres² and A. Falcão²

¹ *Department of Computing, São Paulo State University, Bauru, Brazil*

² *Institute of Computing, University of Campinas, Campinas, Brazil*

Received 14th Dec 2013; accepted 9th Sep 2014

A. Fernandes Mansano / Electronic Letters on Computer Vision and Image

Abstract

In this paper, we deal with the descriptor combination problem in image classification tasks. This problem refers to the definition of an appropriate combination of image content descriptors that characterize different visual properties, such as color, shape and texture. In this paper, we propose to model the descriptor combination as a swarm-based optimization problem, which finds out the set of parameters that maximizes the classification accuracy of the Optimum-Path Forest (OPF) classifier. In our model, a descriptor is seen as a pair composed of a feature extraction algorithm and a suitable distance function. Our strategy here is to combine distance scores defined by different descriptors, as well as to employ them to weight OPF edges, which connect samples in the feature space. An extensive evaluation of several swarm-based optimization techniques was performed. Experimental results have demonstrated the robustness of the proposed combination approach.

Key Words: Image classification, Descriptor Combination, Swarm intelligence, Optimum-Path Forest.

1 Introduction

Image classification covers a broad extent of research fields, which range from medical image analysis to scene understanding in satellite imagery. Given an image, the general pipeline is to apply a preprocessing filter aiming to enhance borders and texture, and then to extract representative features to be used in machine learning systems.

In order to improve image representation, several works have been dedicated to the problem of finding the best image descriptors for different applications. It is known that there is no optimum descriptor, i.e., the one that achieves good effectiveness results for any dataset. In this sense, feature selection approaches are often employed to remove non-relevant features and then to improve the descriptor recognition rates. Another commonly adopted approach relies on the combination of features encoding different visual properties (information fusion). Several works focus on finding the most reliable features, i.e., to select appropriate features according to a predefined criterion [12, 13, 24]. Other works address this problem using Linear Discriminant Analysis [20], Principal Component Analysis [42] and correlation methods [23].

Correspondence to: <papa@fc.unesp.br>

Recommended for acceptance by <Filiberto Pla>

ELCVIA ISSN:1577-5097

Published by Computer Vision Center / Universitat Autònoma de Barcelona, Barcelona, Spain

However, the use of features may not be enough for the construction of effective classification systems. In some cases, the effectiveness performance of features may be improved considerably if more appropriate distance functions are employed. Papa et al. [29], for example, have shown that we can improve the recognition accuracy of a classifier up to 7.28% if we consider to replace the distance function in the context of image classification tasks. Torres et al. [38] have introduced a novel model to represent image descriptors in Content-Based Image Retrieval (CBIR) tasks. According to them, an image descriptor is a pair that encodes a feature vector extraction algorithm along with an appropriate distance function.

Based on this model, Faria et al. [8] proposed a descriptor combination approach using a swarm-based optimization algorithm. This approach was validated in image classification tasks using the Optimum-Path Forest (OPF) classifier [29, 28]. The idea is to model the descriptor combination problem as an optimization task, in which the distances associated with each feature vector were combined by means of a linear function, and the final distance scores were used to weight the edges of the OPF graph. The authors employed the Particle Swarm Optimization (PSO) [16, 32, 10] as the swarm intelligence algorithm. In a more recent work, Mansano et al. [22] extended the above PSO-based combination method by employing a new set of parameters that allows a greater range of possible function combination than just a linear combination. Furthermore, Mansano et al. [22] also introduced the use of the Harmony Search (HS) [11] optimization technique in image classification tasks involving the combination of texture and color information. In both works, the use of swarm-based optimization techniques is motivated by their simplicity and elegance to solve such sort of optimization problems. An interesting work has been done by Silva et al. [35], which employed a similar idea, but in the context of CBIR task.

In this paper, we extended their studies by addressing a wider range of swarm-based optimization techniques for descriptor combination aiming at image classification: Artificial Bee Colony (ABC) [15], Modified ABC (MABC) [1], Bat Algorithm (BA) [43], and Glowworm Swarm Optimization (GSO) [17]. These techniques have been compared against with six variations of the HS algorithm, and also against with three variations of the PSO technique. Therefore, this work compares for the first time fifteen swarm-based techniques for descriptor combination in image classification tasks. The remainder of this paper is organized as follows. Section 2 presents the background theory regarding the descriptor combination model, OPF and the evolutionary-based optimization techniques evaluated in our comparative study. Sections 3 and 4 present the employed approach for descriptor combination and the performed experiments, respectively. Finally, Section 5 states our conclusions.

2 Background Theory

In this section we briefly present the background theory regarding descriptors and OPF classifier, as well as the evolutionary-based optimization techniques employed in this work.

2.1 Descriptor Combination

In this section, we formalize how the proposed image classification system is modeled. Additionally, we present the definitions regarding *simple* and *composite* descriptors, as proposed in [34]. An **image** \hat{I} is a pair (D_I, \vec{I}) , where:

- $D_I \subset \mathbb{Z}^2$ is a finite set of *pixels*, and
- $\vec{I} : D_I \rightarrow D'$ is a function that assigns to each pixel p in D_I a vector $\vec{I}(p)$ of values in some arbitrary space D' (for example, $D' = \mathbb{R}^3$ when a color in the RGB system is assigned to a pixel).

A **simple descriptor** (briefly, **descriptor**) D is defined as a pair (ϵ_D, δ_D) , where:

- $\epsilon_D : \hat{I} \rightarrow \mathbb{R}^n$ is a function, which extracts a *feature vector* $\vec{v}_{\hat{I}}$ from an *image* \hat{I} .

- $\delta_D : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ is a *similarity function* that computes the similarity between two images as the inverse of the distance between their corresponding *feature vectors*.

A **feature vector** $\vec{v}_{\hat{I}}$ of an image \hat{I} is a point in \mathbb{R}^n space: $\vec{v}_{\hat{I}} = (v_1, v_2, \dots, v_n)$, where n is the dimension of the vector.

A **composite descriptor** \hat{D} is a pair $(\mathcal{D}, \delta_{\mathcal{D}})$, where:

- $\mathcal{D} = \{D_1, D_2, \dots, D_k\}$ is a set of k pre-defined simple descriptors.
- $\delta_{\mathcal{D}}$ is a combination function which combines the similarity values obtained from each descriptor $D_i \in \mathcal{D}, i = 1, 2, \dots, k$.

Figure 1 illustrates the above idea regarding simple and composite descriptors.

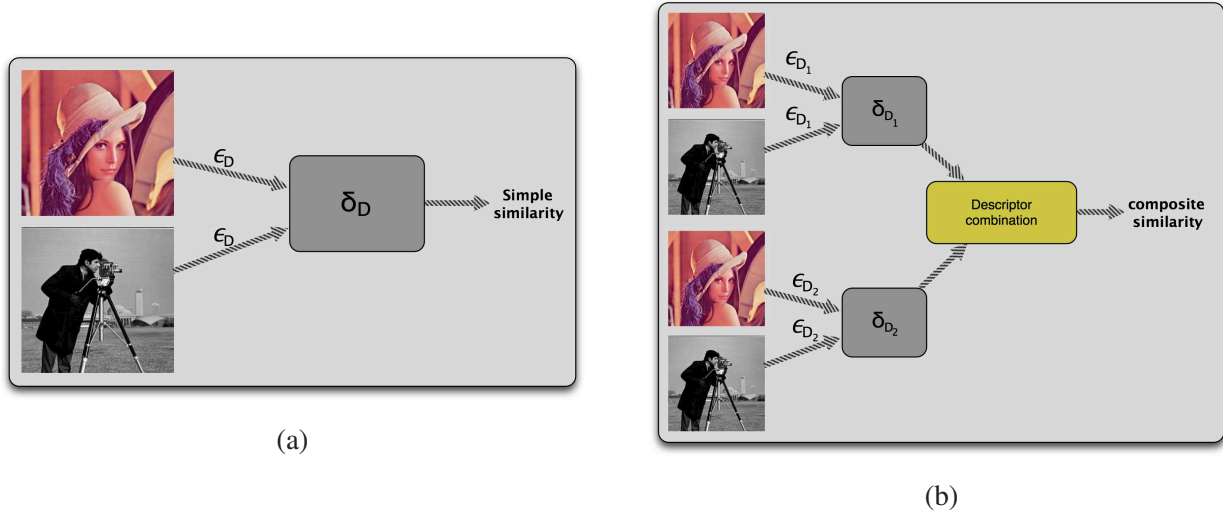


Figure 1: Descriptor combination model using (a) simple and (b) composite descriptors.

2.2 Optimum-Path Forest

The OPF classifier models the problem of pattern recognition as a graph partition task into optimum-path trees (OPTs), being such partitioning process ruled by a competition among some key nodes (prototypes) in order to conquer the remaining samples. Basically, the sample (feature vector) is represented as a graph node, and the edge between two nodes is weighted by a similarity measure between them.

As a community ordered formation process, where groups of individuals are obtained based on optimum connectivity relations to their leaders, OPF employs a competition process among some prototype nodes in order to partition the graph into an optimum-path forest according to some path-cost function. By analogy, the population is divided into communities, where each individual belongs to the group which offered to it the highest reward. Actually, it is possible to understand OPF as a framework for graph-based classification, not only as a sole classifier, which means we just need to change some parameters to obtain a new classifier. The first and widely used version of OPF employs a complete graph as adjacency relation, as well as the prototypes are chosen as the nearest elements from different classes [29, 28]. Another supervised variant of OPF was also proposed by Papa and Falcão [27], which employs a k -nearest neighbours graph as the adjacency relation. In addition, the prototypes are estimated in the regions with highest densities.

The version proposed by Papa et al. [29, 28] elects the prototypes as the nodes that fall in the boundaries of the classes, which work as sentinels to protect samples from a certain class to conquer nodes from other

labels. The prototypes are marked as the connected samples from different classes in a minimum spanning tree (MST) computed over the training set; as the arc-weights encode the distance between samples, the MST algorithm enforces the edges with lowest weights to be part of it, thus connecting samples with minimum distance. Therefore, after computing the prototypes, we can start the competition process using some smooth path-cost function, that is a function that obeys some properties [6]. The path-cost function adopted by the OPF version used in this work is the same as used by Papa et al. [29, 28], which computes the maximum arc-weight along a path between two samples. This path-cost function aims to avoid the well-known chain-code problem, which might degrade the accuracy of OPF if we use a simple summation of the arc-weights as the path-cost function.

The OPF pipeline concerns with a training and a test step: in the training step, the above procedure to find out prototypes and to compute the optimum-path forest are then executed, for further classification of the test nodes. This latter step can be performed by adding each test sample to the optimum-path forest, verifying the training sample that conquer that test node, and then to assign the label of the “winning” node to that test sample. Notice the test node is then removed from the optimum-path forest after its classification.

2.3 Evolutionary Optimization Background

In this section, we briefly review the optimization techniques used in this work.

Particle Swarm Optimization: Particle Swarm Optimization (PSO) is an algorithm modeled on swarm intelligence that finds a solution in a search space based on social behavior dynamics [16]. Each possible solution of the problem is modeled as a particle in the swarm that imitates its neighborhood based on the values of the fitness function found so far.

Other definitions consider PSO as a stochastic and population-based search algorithm, in which social behavior learning allows each possible solution (particle) to “fly” onto this space (swarm) looking for other particles that have better characteristics, i.e., the ones that maximize a fitness function. Each particle has a memory that stores its best solution, as well as the best solution of the entire swarm. Thus, taking this information into account, each particle has the ability to imitate the others that give to it the best local and global maxima. This process simulates the social interaction between humans looking for the same objective, or bird flocks looking for food, for instance. This socio-cognitive mechanism can be summarized into three main principles [16]: (i) evaluation, (ii) comparison, and (iii) imitation. Each particle can evaluate others in its neighborhood through some fitness function, can compare it with its own value and, finally, can decide whether it is a good choice to imitate them. Some of PSO variants can be highlighted: Chaos PSO (CPSO) [19], Elite PSO (EPSO) [39], and PSOW [5]. Basically, such variants try to improve the search for good solutions avoid trappings in local optima solutions.

Harmony Search: Harmony Search (HS) is a meta-heuristic algorithm inspired in the improvisation process of music players where musicians improvise their instruments’ pitches searching for a perfect state of harmony [11]. The main idea is to use the same process adopted by musicians to create new songs aiming to obtain a near-optimal solution according to some fitness function. Each possible solution is modeled as a harmony, and each musical note corresponds to one decision variable.

In the last years, several researches have focused on developing variants based on the original HS proposal [2]. Some of them have proposed different ways to dynamically set HS parameters, while others presented new improvisation schemes. Mahdavi et al. [21], for instance, proposed an Improved Harmony Search (IHS), which differs from the traditional HS as it updates the PAR (Pitch Adjusting Rate) and bandwidth values dynamically.

Later on, Omran and Mahdavi [25] proposed the Global-best Harmony Search (GHS), which has exactly the same steps as IHS, except for the pitch adjustment rule. In their approach, a new solution is proposed by making use of the best harmony vector. Zou et al. [44] proposed the Novel Global Harmony Search (NGHS),

which differs from traditional HS with regard to three aspects: in the NGHS, the HMCR (Harmony Memory Considering Rate) and PAR parameters are excluded; a mutation probability p_m is used; finally, NGHS has a modified improvisation scheme, and it always replaces the worst harmony with the new one. Finally, Pan [26] proposed the Self-adaptive Global best Harmony Search (SGHS) approach, which has a new improvisation scheme based on GHS. In addition, HMCR and PAR are modeled as self-adaptive parameters.

Recently, Yadav et al. [41] have introduced a new HS variant called Intelligent Tuned HS (ITHS), which still depends on parameters used in the original HS, i.e., it does not introduce new parameters as stated by some aforementioned HS-based approaches. Additionally, PAR is automatically computed in ITHS, instead of traditional HS, in which the user needs to input it.

Bat Algorithm: Based on the behavior of bats, Yang [43] has developed a new meta-heuristic optimization technique called Bat Algorithm. Such technique has been developed to behave as a band of bats tracking prey/foods using their capability of echolocation. In order to model this algorithm, Yang [43] has defined some rules, as follows:

1. All bats use echolocation to sense distance, and they also “know” the difference between food/prey and background barriers in some way;
2. A bat flies randomly with a certain velocity at some position with a fixed frequency, varying wavelength and loudness to search for prey. They can automatically adjust the wavelength (or frequency) of their emitted pulses, and adjust the rate of pulse emission, depending on the proximity of their target;
3. Although the loudness can vary in many ways, Yang [43] assumes that the loudness varies from a large (positive) to a minimum constant value.

Artificial Bee Colony: Karaboga and Basturk [15] proposed the Artificial Bee Colony (ABC) optimization algorithm, which is based on the behavior of bees looking for food in order to maintain the hive. The colony consists of three groups of bees: employed bees, onlookers, and scouts. In such model, there is only one artificial employed bee for each food source. Employed bees go to their food source and come back to hive and performs a dance on this area. The employed bee whose food source has been abandoned becomes a scout and starts searching for a new food source. Onlookers watch the dances of employed bees and choose food sources depending on such movements. Further, Akay and Karaboga [1] presented the Modified Artificial Bee Colony (MABC), which contains some modifications in the perturbation process when generating new solutions.

Glowworm Swarm Optimization: The Glowworm Swarm Optimization(GSO) [18] is a swarm intelligence optimization algorithm based on the behavior of glowworms. The main idea is to use the capability of glowworms to change the intensity of the luciferin emission and thus appear to glow at different intensities. Therefore, the light intensity will be used as the fitness function to attract other glowworms, being their position the possible solution of the optimization model.

3 Optimization-based Descriptor Combination

Faria et al. [8] have proposed an optimization-based approach for descriptor combination where the best descriptor was the one that maximized the accuracy of OPF classifier in an evaluating set. Let $\hat{D}^* = (\mathcal{D}^*, \delta_{\mathcal{D}}^*)$ be such descriptor. We have that $\delta_{\mathcal{D}}^*$ can be formulated as:

$$\delta_{\mathcal{D}}^* = \sum_{i=1}^N \alpha_i \delta_{\mathcal{D}_i}, \quad (1)$$

in which $\alpha_i \in \mathbb{R}$ stands for the parameter to be optimized and δ_{D_i} denotes the distance function for descriptor \hat{D}_i . Thus, the descriptor combination task means to combine the distances of each descriptor and to use them to weight the arcs between samples in the OPF graph. In a practical perspective, δ_{D_i} corresponds to a square matrix that stores the distances between all pairs of training nodes.

Let $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_N)$ be the set of all parameters. One can observe that δ_D^* is a linear combination of all distance matrices, and α corresponds to the set of parameters that will be optimized by some approach. In that case, Faria et al. [8] employed PSO for such task, and their idea was to use the OPF recognition rate over an evaluating set as the function to be maximized.

Further, Mansano et al. [22] extended the above formulation beyond a linear combination of α by introducing a new set of parameters $\beta = (\beta_1, \beta_2, \dots, \beta_N)$. Therefore, Equation 1 is rewritten as:

$$\delta_D^* = \sum_{i=1}^N \alpha_i \delta_{D_i}^{\beta_i}, \quad (2)$$

in which $-2 \leq \alpha_i, \beta_i \leq 2$, $\beta_i \in \mathbb{R}$. The main idea behind such formulation is to allow a greater variability of arithmetic computations, which have been restricted to only the α set in Equation 1. The range of values for α_i and β_i , $i = 1, 2, \dots, N$, have been empirically set.

Let $\xi = \{(\alpha_i, \beta_i)\}$, $i = 1, 2, \dots, N$, be the set of parameters to be optimized, and $\xi^* = (\alpha_i^*, \beta_i^*)$, $i = 1, 2, \dots, N$, be the one that maximizes the OPF accuracy over the evaluation set. In order to make it clear, suppose we have a dataset with two descriptors extracted. Thus, $\xi = \{(\alpha_1, \beta_1); (\alpha_2, \beta_2)\}$. In this case, the optimization technique search space would have four dimensions, and each swarm agent (particle, harmony, bee, bat, etc) position j would be modeled by $\vec{x}_j = (\alpha_1^j, \alpha_2^j, \beta_1^j, \beta_2^j)$, $1 \leq j \leq M$. In this case, M stands for the number of agents.

The methodology proposed here has two phases [22]: (i) the *swarm-based optimization phase* which employs a training and evaluation sets to find ξ^* , (ii) and the *classification phase* that computes the new distance d for each test sample s to all training samples, as stated in the OPF classification procedure [29]. Let $d(s, x)$ be the distance between s and the training sample x . Thus, $d(s, x)$ can be found as follows:

$$d(s, x) = \sum_{i=1}^N \alpha_i^* \delta_{D_i}^{\beta_i^*}(s, x), \quad (3)$$

in which $\delta_{D_i}(s, x)$ denotes the distance between samples s and x using descriptor i . Therefore, the idea is to model the task description combination as an optimization problem, in which the combined distances will weight the arcs in the OPF classification schema. Figure 2 depicts the above descriptor combination model based on swarm intelligence.

The set of parameters ξ is then obtained over the training and evaluation sets, being this latter employed to avoid overfitting in the training set. As such, the evaluation set may encode the properties of the unseen test set, which represents the real-world scenario. As depicted in Figure 2, each agent encodes the possible set of values for each parameter to be optimized, which are then used to map the training and evaluation sets to a new representation considering these parameters. Further, the OPF classifier is trained with this new set and its effectiveness is assessed over the new evaluation set. The accuracy over this set is then used as the fitness function to be maximized. This procedure is carried out for all aforementioned swarm-based optimization techniques until a stop criterion is reached. Then, the set of parameters ξ^* is finally used to map the unseen test set to a new representation for recognition rates computation using Equation 3.

4 Experimental Results

In this section, we present the experiments conducted to compare the swarm-based optimization techniques used for descriptor combination. In order to accomplish this task, we use two well-known public datasets, as

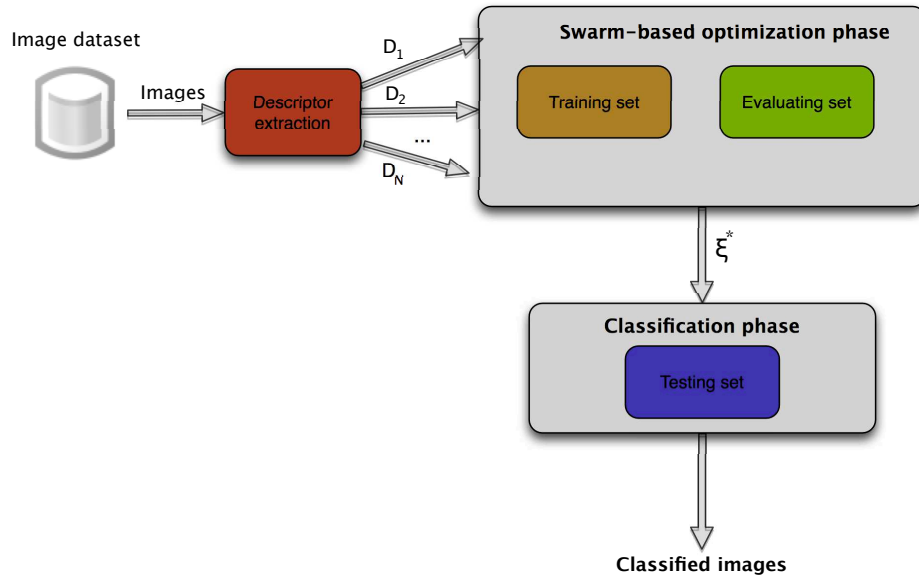


Figure 2: Swarm-based descriptor combination model.

follows:

- Corel*: this dataset contains 3,906 images labeled in 85 classes, and the number of images per class ranges from 7 to 98 images; and
- Free Foto[†]: we used a subset containing 3,426 images labeled in 9 classes, and the number of images per class ranges from 70 to 854 images.

In regard to the feature extraction step, we employed Local Activity Spectrum (LAS), Global Color Histogram (GCH), Color Autocorrelogram (ACC), Border/Interior pixel Classification (BIC), and Quantized Compound Change Histogram (QCCH) for the Corel dataset; and LAS, GCH, Homogeneous Texture Descriptor (HTD), BIC, and Color Coherent Vector (CCV) for the Freefoto dataset:

- Local Activity Spectrum: texture descriptor in which components are quantized in 4 bins resulting in a histogram of 256 bins [37].
- Global Color Histogram: the color histogram models the color distribution in an image. Roughly speaking, the GCH steps concern with quantizing the image for further histogram computation [37]. The obtained histogram can be used to represent the entire image, being a matching function used to compute the similarity between two histograms.
- Color Autocorrelogram: this feature extraction algorithm maps the probability of pixels with different colours to be in a certain distance to each other, being such distance a parameter of this approach. One common approach to measure the distance between two ACC-based feature vectors is to employ the well-known L1 norm [37].
- Border/Interior pixel Classification: the idea of BIC is to classify each image's pixel as being a border or interior, and then to use a logarithmic-based distance to compare different histograms [36]. It is a fast and very accurate image description technique.

*<http://vision.stanford.edu> (as of May, 2014).

[†]<http://www.freefoto.com> (as of May, 2014)

- Quantized Compound Change Histogram: it is a texture descriptor composed by 40 bins, which aims to detect the compound rate of grey change for each pixel in four different regions in its neighborhood [14].
- Color Coherence Vector: this technique aims at classifying each pixel as being coherent or incoherent, being coherent pixels the ones that belong to some big connected component, while incoherent pixels are part of small connected components [30].
- Homogeneous Texture Descriptor: this feature extraction algorithm is composed by 64 integer values from the Gabor filter response of 30 frequency channels [40].

In this work, we employed different image descriptors for each dataset due to the good results achieved in two previous experiments [9, 7]. A detailed description about these descriptors can be found in [31]. Recall that all experiments were carried out with the OPF classifier and Euclidean distance for both simple and composite descriptors. In order to assess the improvement in regard to the proposed descriptor combination approach, we have initially executed experiments with the simple descriptors over 5 rounds with cross-validation. For that, 50% of the entire dataset was used for training and 50% for classification purposes. Notice the accuracy used was the one proposed by Papa et al. [29], which considers unbalanced data: let $|Z_i|$ - $i = 1, 2, \dots, c$ - be the number of samples of a dataset Z from each class i and c , the number of classes. We can define

$$e_{i,1} = \frac{FP_i}{|Z| - |Z_i|} \text{ and } e_{i,2} = \frac{FN_i}{|Z_i|}, \quad i = 1, \dots, c, \quad (4)$$

where $FP(i)$ and $FN(i)$ are the false positives and false negatives, respectively. That is, FP_i is the number of samples from other classes that were classified as being from the class i in Z , and FN_i is the number of samples from class i that were incorrectly classified as being from other classes in Z .

The errors $e_{i,1}$ and $e_{i,2}$ are used to define

$$E_i = e_{i,1} + e_{i,2}, \quad (5)$$

where $E(i)$ is the partial sum error of class i . Finally, the accuracy is written as

$$Acc = \frac{2c - \sum_{i=1}^c E_i}{2c} = 1 - \frac{\sum_{i=1}^c E_i}{2c}. \quad (6)$$

Figure 3 displays the OPF mean accuracy for each simple descriptor. We can see the most effective descriptors have been ACC and BIC for Corel and Freefoto datasets, respectively: ACC obtained $73.51\% \pm 0.4$ of recognition rate, while BIC achieved $92.99\% \pm 0.49$, being 8.48% more accurate than the second best descriptor (CCV) for Freefoto dataset. The main problem of having such a very good descriptor with respect to the others is related to the fact it can neglect the descriptor combination process, which will be biased on this descriptor. The next experiments focus on this statement.

The experiments involving descriptor combination employed the same training and test sets as before with cross-validation over 5 rounds. However, we use now an evaluation set with 20% of the entire dataset size, which was obtained from the training set used in the previous experiment. As aforementioned, the OPF recognition rate over such set will be used as the fitness function by the swarm-based optimization techniques. In this work, we have employed the following optimization approaches:

- PSO and three variants: Chaos PSO (CPSO) [19], Elite PSO (EPSO) [39], and PSOW [5];
- HS and six variants: Chaos HS (CHS) [3], Mahdavi HS (MHS) [21]; Global-best HS (GHS) [25], Self-adaptative GSH (SGHS) [33], Intelligent Tuned HS (ITHS) [41]; and Novel Global HS (NGHS) [4];
- Remaining approaches: BA [43], ABC [15], MABC [1], and GSO [17].

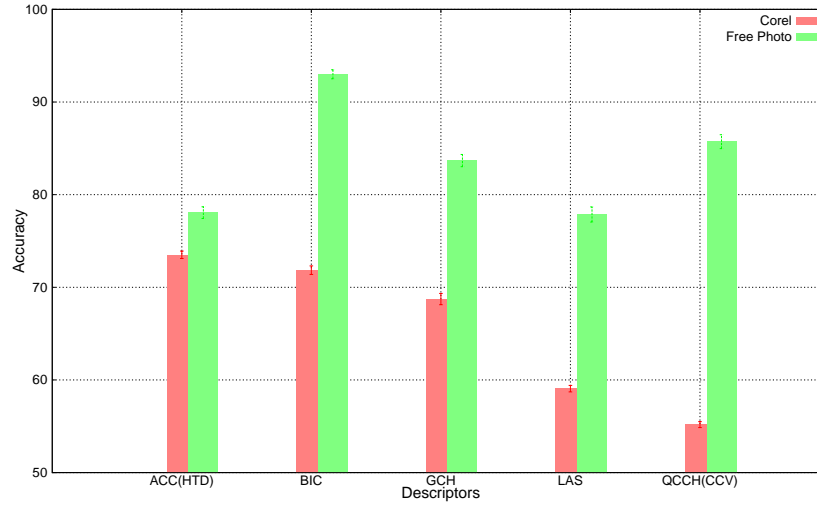


Figure 3: Simple descriptor effectiveness using OPF for the Corel and Freefoto datasets.

Table 1 displays the parameter values used for each technique, which were empirically set. In regard to PSO and its variants, w stands for the inertia weight, while w_i and w_f denote the initial and final values of w , respectively. Notice that PSOW, EPSO, and CPSO employ a different value for inertia during the convergence process (standard PSO uses the same value w for all iterations); CLS means the Chaos Local Search number iterations, which stands for a local search performed by CPSO. With regard to HS and its variants, we have the Harmony Memory Considering Rate ($HMCR$), Pitch Adjusting Rate (PAR), maximum and minimum PAR values represented by PAR_{min} and PAR_{max} , respectively; b_{min} and b_{max} that denote the minimum and maximum bandwidth values, respectively; and p_m concerns with the mutation probability. Considering ABC and MABC, $limit$ is an *ad-hoc* parameter, and MR stands for the modification rate. The parameters β and γ for BA and GSO are *ad-hoc* constants; L_0 and R_0 stand for the initial amount of luciferin and its decay, respectively; $STEP$ means the step size used for each glowworm towards the direction of other ones; and NT and RS denote the glowworm's neighborhood size and its radius, respectively. Recall that for all approaches we have employed 250 agents (particles for PSO, harmonies for HS, bats for BA, bees for ABC and glowworm for GSO) and 100 iterations for convergence. In case of ABC and MABC, 125 bees are used in the scouting phase, and the remaining 125 bees are used in the exploitation step.

We have conducted two different descriptor combination experiments: LAS+GCH and ALL. While the former allows us to combine texture (LAS) and color (GCH) information, in the latter we opted to use all available descriptors. The initial choice for LAS and GCH in the first experiment is based on two reasons: first, they encode information of two different visual properties (color and texture); second, they usually are not correlated, providing complementary information regarding image content [31].

Figure 4 shows the results for LAS+GCH experiment. In regard to Free Foto dataset, CHS was the best approach with $84.5\% \pm 0.82$ of recognition rate, while the best individual descriptor was GCH with $83.68\% \pm 0.66$ of classification rate, which means both techniques are statistically similar to each other. Considering Corel dataset, the best individual descriptor was GCH with $68.73\% \pm 0.61$ of recognition rate, while PSOW was the best combination technique with $69.5\% \pm 0.35$. Therefore, we can conclude both techniques are statistically similar. The idea of this experiment is just to show a situation in which descriptor combination may not substantially improve the results when we have a descriptor that is much better than others: in this case, GCH was 16.37% more accurate than LAS for Corel dataset, and 7.47% more effective than LAS concerning Free Foto dataset.

Figure 5 displays the computational load [s] considering the experiment LAS+GCH. Techniques based on HS have been the most efficient ones, since they do not update all agents in a single iteration, just the worst one. Usually, swarm-based techniques update all agents, which may be time-consuming, but faster for convergence.

Table 1: Parameters used for each optimization technique.

Technique	Parameters
PSO	$c_1 = 1.6, c_2 = 0.4, w = 0.7$
PSOw	$c_1 = c_2 = 2, w_i = 0.9, w_f = 0.7$
EPSO	$c_1 = c_2 = 2, w_i = 0.9, w_f = 0.7$
CPSO	$c_1 = c_2 = 2, w_i = 0.9, w_f = 0.7, CLS = 30$
HS	$PAR = HMCR = 0.7$
CHS	$PAR_{min} = 0.1, PAR_{max} = 1.0, HMCR = 0.7$
MHS	$PAR_{min} = 0.1, PAR_{max} = 1.0, HMCR = 0.7$ $b_{min} = 0.01, b_{max} = 0.25$
NGHS	$PAR = HMCR = 0.7, p_m = 0.7$
SGHS	$PAR_{min} = 0.1, PAR_{max} = 1.0, HMCR = 0.7$
GHS	$PAR_{min} = 0.1, PAR_{max} = 1.0, HMCR = 0.7$
ITHS	$PAR_{min} = 0.1, PAR_{max} = 1.0, HMCR = 0.7$
GSO	$\beta = 0.08, \gamma = 0.6, L_0 = 5, R_0 = 4, STEP = 3$ $NT = 5, RS = 1$
ABC	$limit = 50, MR = 0.5$
MABC	$limit = 50, MR = 0.5$
BA	$\beta = 0.9, \gamma = 3$

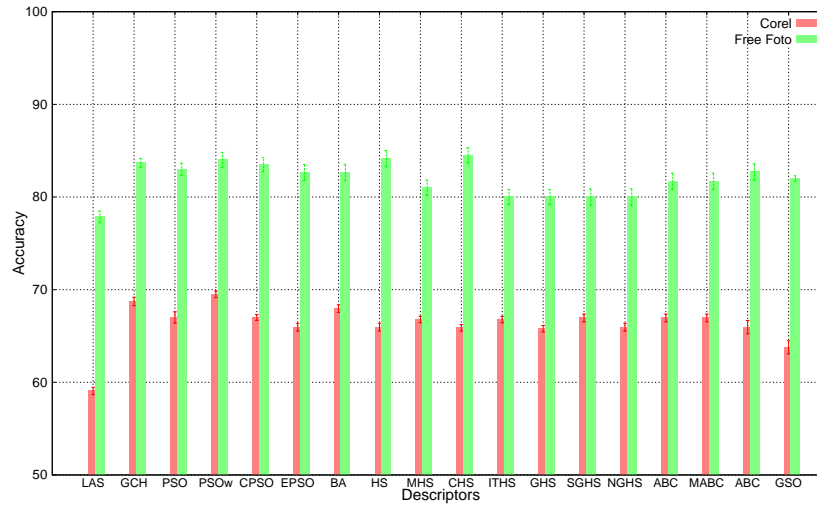


Figure 4: Composite descriptor effectiveness using OPF and LAS+GCH descriptors.

In regard to Corel dataset, for instance, SGHS was 232 times faster than the best approach PSOw. This might be a considerable improvement in time, since PSOw was only 3.8% more accurate than SGHS in this dataset. Therefore, if one can seeks for a good trade-off between effectiveness and efficiency, HS-based techniques seem to be a good choice for that.

Figure 6 depicts the results concerning ALL experiment. The best results considering Free Foto dataset were obtained by EPSO with $93.64\% \pm 0.89$ of recognition rate, as well as by BIC single descriptor with $92.99\% \pm 0.49$ of classification accuracy. A new descriptor called “Raw” has been added to the experimental section, which consists in a concatenation of all individual descriptors. Such technique obtained $87.64\% \pm 0.89$ of recognition rate, outperforming all single descriptors except BIC. We have used “Raw” as our baseline for descriptor combination techniques. For this dataset, both EPSO and BIC are statistically similar to each other.

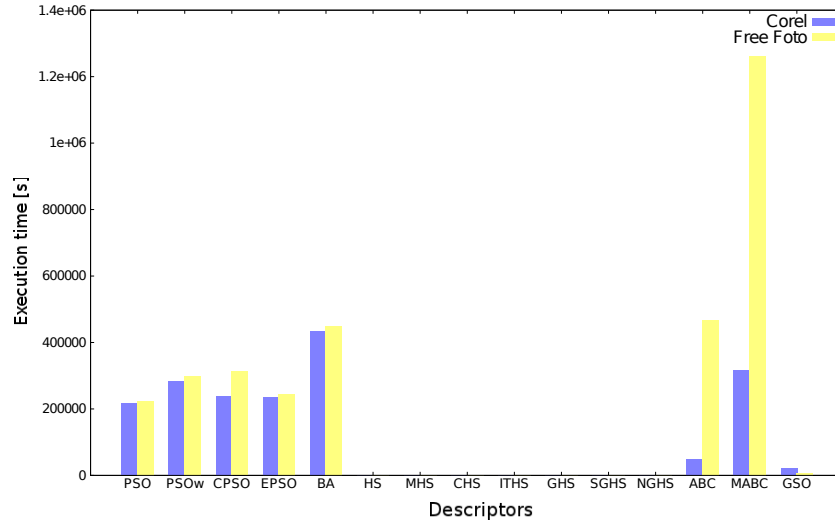


Figure 5: Composite descriptor efficiency using OPF and LAS+GCH descriptors.

However, if we consider Corel dataset, an improvement obtained by descriptor combination can be highlighted. BA and MABC obtained the best results with exact the same recognition rate of $77.95\% \pm 0.42$, being the best individual descriptor ACC with $73.51\% \pm 0.40$ of classification accuracy. In this dataset, there is no evidence that a single descriptor substantially outperforms others, which is a very good situation for descriptor combination, as stated before. In regard to computational load, HS-based approaches have been the fastest ones, as can be evidenced in Figure 7. The behavior concerning efficiency for ALL experiment follows the same as in LAS+GCH one, except for ABC-based approaches, which are amongst the slowest ones.

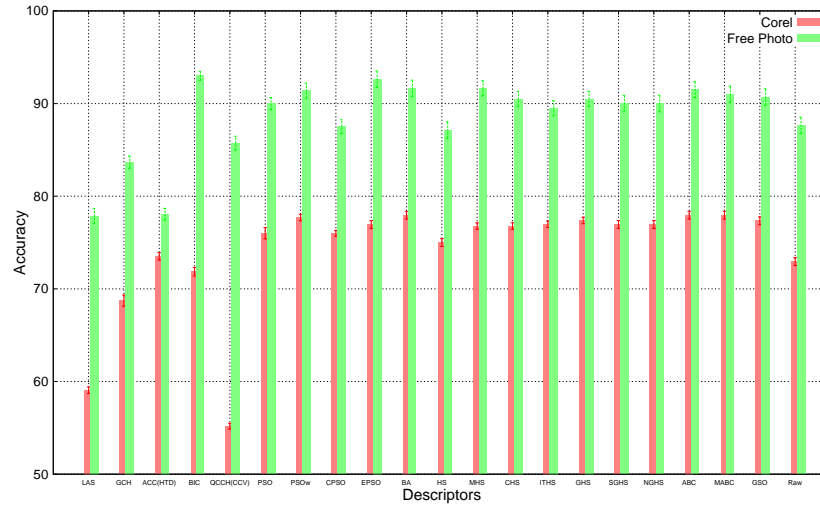


Figure 6: Composite descriptor effectiveness using OPF and ALL descriptors.

5 Conclusions

Image classification has been actively pursued in the last years, mainly because the wide range of applications that consider an image as input. In this context, combining information provided by different image description approaches may improve the performance of classification systems. In this paper, we have addressed a

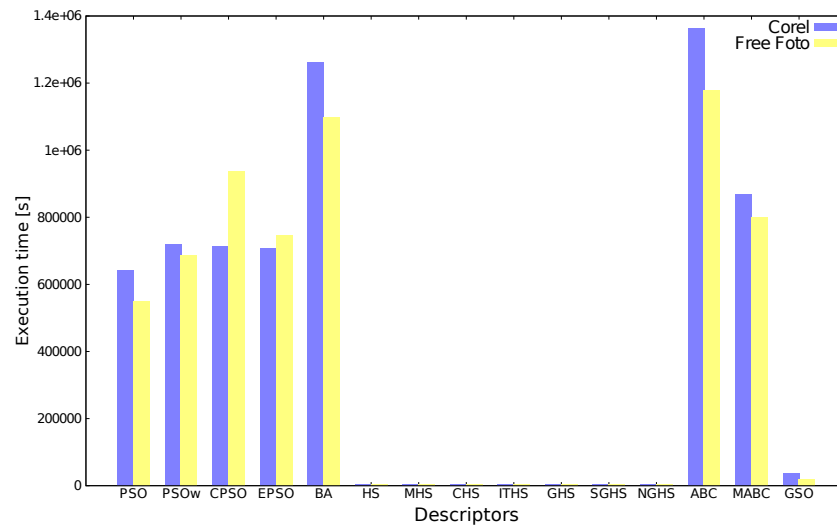


Figure 7: Composite descriptor efficiency using OPF and ALL descriptors.

workaround for such problem, in which the descriptor combination is modeled as an optimization problem, and the fitness function to be maximized is the OPF accuracy over an evaluating set.

We have compared fifteen swarm-based optimization techniques in two public datasets and two different descriptor combination experiments, using only two descriptors and also all of them. Although all techniques have achieved close results, HS-based approaches are more suitable for such task, since they can provide good results being faster than all other methods. We also highlight the descriptor combination approach proposed in this paper can improve the effectiveness of single descriptors. However, such improvement may not improve a lot the efficiency of a very good single descriptor, since this descriptor tends to dominate the other ones, forcing the optimization techniques to converge faster.

In regard to the computational load, the process of learning parameters might be done off-line, since it can be time consuming, mainly for large datasets. Additionally, the experiments shed light over the importance of combining different image content properties on image classification tasks.

Acknowledgments

The authors are grateful to AMD, CAPES, CNPq (grants #306580/2012-8, #484254/2012-0, #470571/2013-6, #303182/2011-3, #303673/2010-9, #479070/2013-0 and #1260-12-0), FAPESP (grants #2010/14910-0 and #2013/20387-7), and Microsoft-FAPESP Virtual Institute (grant #2013/50169-1).

References

- [1] Akay, B., Karaboga, D.: A modified artificial bee colony algorithm for real-parameter optimization. *Information Sciences* **192**, 120–142 (2012)
- [2] Alia, O., Mandava, R.: The variants of the harmony search algorithm: an overview. *Artificial Intelligence Review* **36**, 49–68 (2011)
- [3] Coelho, L.S., Bernert, D.L.A.: An improved harmony search algorithm for synchronization of discrete-time chaotic systems. *Chaos, Solitons and Fractals* **41**, 830–848 (2009)
- [4] D. Zou L. Gao, J.W., Li, S.: Novel global harmony search algorithm for unconstrained problems. *Neurocomputing* **73**, 3308–3318 (2010)

- [5] Eberhart, R.C., Shi, Y.: Comparing inertia weights and constriction factors in particle swarm optimization. In: *Proceedings of the 2000 Congress on Evolutionary Computation*, 2000, pp. 84–88 (2000)
- [6] Falcão, A., Stolfi, J., Lotufo, R.: The image foresting transform theory, algorithms, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **26**(1), 19–29 (2004)
- [7] Faria, F.A.: *Uso de Técnicas de Aprendizagem para Classificação e Recuperação de Imagens*. Master's thesis, Institute of Computing, University of Campinas, Brazil (2010)
- [8] Faria, F.A., Papa, J.P., Torres, R.S., Falcão, A.: Multimodal pattern recognition through particle swarm optimization. In: *Proceedings of the 17th International Conference on Systems, Signals and Image Processing*, pp. 1–4. Rio de Janeiro, Brazil (2010)
- [9] Faria, F.A., Veloso, A., Almeida, H.M., Valle, E., Torres, R.d.S., Gonçalves, M.A., Meira Jr., W.: Learning to rank for content-based image retrieval. In: *Proceedings of the international conference on Multimedia information retrieval, MIR '10*, pp. 285–294. ACM, New York, NY, USA (2010)
- [10] Fernández Martínez, J., García Gonzalo, E.: The PSO family: deduction, stochastic analysis and comparison. *Swarm Intelligence* **3**(4), 245–273 (2009)
- [11] Geem, Z.W.: *Music-Inspired Harmony Search Algorithm: Theory and Applications*, 1st edn. Springer Publishing Company, Incorporated (2009)
- [12] Gehler, P., Nowozin, S.: On feature combination for multiclass object classification. In: *Proceedings of the 12th International Conference on Computer Vision*, pp. 221–228 (2009)
- [13] Hou, J., Zhang, B.P., Qi, N.M., Yang, Y.: Evaluating feature combination in object classification. In: *Proceedings of the 7th International Conference on Visual Computing*, pp. 597–606. Springer-Verlag, Las Vegas, NV, USA (2011)
- [14] Huang, C.B., Liu, Q.: An orientation independent texture descriptor for image retrieval. In: *Proceedings of the International Conference on Communications, Circuits and Systems*, pp. 772–776 (2007)
- [15] Karaboga, D., Basturk, B.: A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization* **39**(3), 459–471 (2007)
- [16] Kennedy, J., Eberhart, R.: *Swarm Intelligence*. M. Kaufman (2001)
- [17] Krishnanand, K., Ghose, D.: Glowworm swarm optimization for simultaneous capture of multiple local optima of multimodal functions. *Swarm Intelligence* **3**, 87–124 (2009)
- [18] Krishnanand, K.N., Ghose, D.: Glowworm swarm based optimization algorithm for multimodal functions with collective robotics applications. *Multiagent and Grid Systems* **2**, 209–222 (2006)
- [19] Liu, B., Wang, L., Jin, Y.H., Tang, F., Huang, D.X.: Improved particle swarm optimization combined with chaos. *Chaos, Solitons; Fractals* **25**(5), 1261–1271 (2005)
- [20] Liu, X., Zhang, L., Li, M., Zhang, H., D.Wang: Boosting image classification with LDA-based feature combination for digital photograph management. *Pattern Recognition* **38**(6), 887–901 (2005)
- [21] Mahdavi, M., Fesanghary, M., Damangir, E.: An improved harmony search algorithm for solving optimization problems. *Applied Mathematics and Computation* **188**(2), 1567 – 1579 (2007)
- [22] Mansano, A.F., Matsuoka, J.A., Afonso, L.C.S., Papa, J.P., Faria, F.A., Torres, R.d.S.: Improving image classification through descriptor combination. In: *Proceedings of the 25th SIBGRAPI Conference on Graphics, Patterns and Images*, pp. 324–329 (2012)

- [23] Neumayer, R., Nørnvåg, K.: Evaluation of feature combination approaches for text categorisation. In: Proceedings of the 19th international conference on Foundations of intelligent systems, ISMIS'11, pp. 438–448. Springer-Verlag, Berlin, Heidelberg (2011)
- [24] Okanohara, D., Tsujii, J.: Learning combination features with L1 regularization. In: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the ACL, pp. 97–100. Stroudsburg, PA, USA (2009)
- [25] Omran, M.G., Mahdavi, M.: Global-best harmony search. *Applied Mathematics and Computation* **198**(2), 643 – 656 (2008)
- [26] Pan, Q.K., Suganthan, P., Tasgetiren, M.F., Liang, J.: A self-adaptive global best harmony search algorithm for continuous optimization problems. *Applied Mathematics and Computation* **216**(3), 830 – 848 (2010)
- [27] Papa, J.P., Falcão, A.X.: A new variant of the optimum-path forest classifier. In: Proceedings of the 4th International Symposium on Advances in Visual Computing, pp. 935–944. Springer-Verlag, Berlin, Heidelberg (2008)
- [28] Papa, J.P., Falcão, A.X., Albuquerque, V.H.C., Tavares, J.M.R.S.: Efficient supervised optimum-path forest classification for large datasets. *Pattern Recognition* **45**(1), 512–520 (2012)
- [29] Papa, J.P., Falcão, A.X., Suzuki, C.T.N.: Supervised pattern classification based on optimum-path forest. *International Journal of Imaging Systems and Technology* **19**(2), 120–131 (2009)
- [30] Pass, G., Zabih, R., Miller, J.: Comparing images using color coherence vectors. In: Proceedings of the Fourth ACM International Conference on Multimedia, pp. 65–73. ACM, New York, NY, USA (1996)
- [31] Penatti, O., Valle, E., Torres, R.: Comparative study of global color and texture descriptors for web image retrieval. *Journal of Visual Communication and Image Representation* **23**(2), 359–380 (2012)
- [32] Poli, R., Kennedy, J., Blackwell, T.: Particle swarm optimization. *Swarm Intelligence* **1**(1), 33–57 (2007)
- [33] Q. Pan P. N. Suganthan, M.F.T., Liang, J.: A self-adaptative global best harmony search algorithm for continuous optimization problems. *Applied Mathematics and Computation* **216** (2010)
- [34] da S. Torres, R., Falcão, A.X.: Content-based image retrieval: Theory and applications. *Revista de Informática Teórica e Aplicada (RITA)* **13**(2), 161–185 (2006)
- [35] da Silva, A.T., dos Santos, J.A., Falcão, A.X., da S. Torres, R., Magalhães, L.P.: Incorporating multiple distance spaces in optimum-path forest classification to improve feedback-based learning. *Computer Vision and Image Understanding* **116**(4), 510–523 (2012)
- [36] Stehling, R.O., Nascimento, M.A., Falcão, A.X.: A compact and efficient image retrieval approach based on border/interior pixel classification. In: Proceedings of the Eleventh International Conference on Information and Knowledge Management, pp. 102–109. ACM, New York, NY, USA (2002)
- [37] Stockman, G., Shapiro, L.G.: *Computer Vision*, 1st edn. Prentice Hall PTR, Upper Saddle River, NJ, USA (2001)
- [38] Torres, R., Falcão, A., Gonçalves, M., Papa, J., Zhang, B., Fan, W., Fox, W.: A genetic programming framework for content-based image retrieval. *Pattern Recognition* **42**(2), 283–292 (2009)
- [39] Wei, J., Guangbin, L., Dong, L.: Elite particle swarm optimization with mutation. In: *System Simulation and Scientific Computing, 2008. ICSC 2008. Asia Simulation Conference - 7th International Conference on*, pp. 800–803 (2008)

- [40] Wu, P., Ro, Y.M., Won, C.S., Choi, Y.: Texture descriptors in MPEG-7. In: W. Skarbek (ed.) *Computer Analysis of Images and Patterns, Lecture Notes in Computer Science*, vol. 2124, pp. 21–28. Springer Berlin Heidelberg (2001)
- [41] Yadav, P., Kumar, R., Panda, S., Chang, C.: An intelligent tuned harmony search algorithm for optimisation. *Information Sciences* **196**, 47–72 (2012)
- [42] Yan, F., Yanming, X.: Image classification based on multi-feature combination and PCA-RBaggSVM. In: *Proceedings of the IEEE International Conference on Progress in Informatics and Computing*, vol. 2, pp. 888–891 (2010)
- [43] Yang., X.S.: Bat algorithm for multi-objective optimisation. *International Journal of Bio-Inspired Computation* **3**(5), 267–274 (2011)
- [44] Zou, D., Gao, L., Wu, J., Li, S.: Novel global harmony search algorithm for unconstrained problems. *Neurocomputing* **73**(1618), 3308 – 3318 (2010)